Analysis of Computationally Demanding Models with Continuous and Categorical Inputs

Curtis B. Storlie[a], Brian J. Reich[b], Jon C. Helton[c], Laura P. Swiler[d], Cedric J. Sallaberry[d]

[a]Los Alamos National Laboratory, NM,

[b]North Carolina State University, Raleigh, NC

[c]Arizona State University, Tempe, AZ

[d]Sandia National Laboratories, Albuquerque, NM

February 20, 2012

Abstract

The analysis of many physical and engineering problems involves running complex computational models (simulation models, computer codes). With problems of this type, it is important to understand the relationships between the input variables (whose values are often imprecisely known) and the output, and to characterize the uncertainty in the output. Often, some of the input variables are categorical in nature (e.g. pointer variables to alternative models or different types of material, etc.). A computational model that sufficiently represents reality is often very costly in terms of run time. When the models are computationally demanding, meta-model approaches to their analysis have been shown to be very useful. However, the most popular meta-models for computational computer models do not explicitly allow for categorical input variables. In this case, categorical inputs are simply ordered in some way and treated as continuous variables in the estimation of a meta-model. In many cases, this can lead to undesirable and misleading results. In this paper, two meta-models based on functional ANOVA decomposition are presented that explicitly allow for an appropriate treatment of categorical inputs. The effectiveness of the presented meta-models in the analysis of models with continuous and categorical inputs is illustrated with several test cases and also with results from a real analysis.

Keywords: Categorical Inputs; Meta-model; Surrogate model; Emulator; Nonparametric regression; Gaussian Process; Sensitivity analysis; Uncertainty analysis.

Running title: Categorical Input Meta-Models

Corresponding Author: Curtis Storlie, storlie@lanl.gov

1 Introduction

The analysis of many physical and engineering problems involves running complex computational models (simulation models, computer codes). With problems of this type, it is important to understand the relationships between the input variables (whose values are often imprecisely known) and the output. Often, some of the input variables are categorical in nature (e.g. pointer variables to alternative models or discrete design choices, etc.). A computational model that sufficiently represents reality is often very costly in terms of run time. When the models are computationally demanding, meta-model approaches to their analysis have been shown to be very useful [1, 2, 3, 4]. However, the meta-models most commonly used thus far for this purpose do not explicitly allow for categorical input variables. Hence, these categorical inputs must then be ordered in some way and treated as continuous variables when estimating a meta-model. In some cases, this can lead to undesirable and misleading results. There is clearly an intuitive appeal for a more appropriate treatment of categorical inputs than simply treating categorical inputs as continuous variables. In this article, we review some of the existing approaches to this problem [5, 6] and present two new approaches based on Smoothing Spline ANOVA decomposition. In addition, we demonstrate the practical benefits to a more appropriate treatment of categorical variables when performing uncertainty analysis (UA) and sensitivity analysis (SA) with several examples.

In general, we will consider complex computer models of the form

$$\boldsymbol{y} = f(\boldsymbol{x}) + \boldsymbol{\varepsilon}, \tag{1.1}$$

where $\boldsymbol{y} = (y_1, \ldots, y_m)$ is a vector of outputs, $\boldsymbol{x} = [x_1, x_2, \ldots, x_p]$ is a vector of imprecisely known inputs, and $\boldsymbol{\varepsilon}$ is a vector of errors (usually small) incurred by the numerical method used to solve for $\boldsymbol{y}$ [7, 4]. Although analyses for real systems almost always involve multiple output variables as indicated above, the following discussions assume that a single real-valued result of the form $y = f(\boldsymbol{x}) + \varepsilon$ is under consideration. This simplifies the notation and the results under discussion are valid for individual elements of $\boldsymbol{y}$. In the case of many outputs, it may be beneficial to reduce the dimensionality with principle components or wavelet decomposition as presented in Higdon et al. [8] and Bayarri et al. [9], respectively.

The model f can be quite large and involved (e.g., a system of nonlinear partial differential equations requiring numerical solution or possibly a sequence of complex, linked models as is the case in a probabilistic risk assessment for a nuclear power plant [10] or a performance assessment for a radioactive waste disposal facility [11]); the vector $\boldsymbol{x}$ of analysis inputs can be of high dimension and complex structure (e.g., several hundred variables, with individual variables corresponding to physical properties of the system under study or perhaps to designators for alternative models). The uncertainty in each element of $\boldsymbol{x}$ is typically characterized by a probability distribution. Such distributions are intended to numerically capture the existing knowledge about the elements of $\boldsymbol{x}$ and are often developed through an expert review process or based on experimental data; see [12, 13] for more on the characterization of input variable uncertainty.

The uncertainty in the input vector $\boldsymbol{x}$ creates two important components to the analysis of these complex models (i) sensitivity analysis (SA) and (ii) uncertainty analysis (UA). The purpose of SA is to identify the most significant factors or variables affecting the model predictions. The purpose of UA is to quantify the uncertainty in the model results due to the uncertainty in the inputs. If experimental data is available, there is also the problem of model calibration/validation. We do not consider this problem here, but refer the reader to [2, 14] for more details. There are a number of approaches to SA, including differential analysis, variance decomposition procedures, Monte Carlo (sampling-based) analysis, and response surface methods [12, 13, 15]. UA is usually performed by estimating quantities such as threshold exceedence probabilities, quantiles of y, or even the entire CDF of y.

In this paper, we consider the problem of performing SA and UA on complex models with mixed continuous/categorical inputs. We propose two meta-models that explicitly allow for appropriate treatment of categorical inputs. In addition to the intuitive appeal of this treatment, we demonstrate the practical benefits with test cases and and also with results from an actual analysis. The rest of the paper is organized as follows. In Section 2, we describe two new meta-model approaches which explicitly allow for categorical variables and review an extension to the popular Gaussian Process (GP) meta-model which allows for categorical inputs. We then describe the use of these meta-models for the purpose of SA and UA in Section 3. A simulation study to illustrate the properties of the proposed methodology is described in Section 4. In Section 5 we demonstrate the utility of these methods with results obtained in an analysis for the proposed Yucca Mountain repository for high-level radioactive waste [16, 17]. Finally, a concluding discussion is given in Section 6.

2 Meta-Models allowing for Categorical Inputs

In this section, we first review an extension of the popular Gaussian Process (GP) meta-model put forth by [5, 6]. We then describe an extension to categorical inputs for two new meta-model approaches [18, 19] that account for categorical inputs and reduce the dimensionality of the problem through functional ANOVA and variable selection.

2.1 Gaussian Process Models for Nonparametric Regression

Gaussian Process (GP) models have gained much popularity for use as a meta-model in computer experiments; see [1, 2, 20, 21]. Most of this previous work assumes that all inputs are continuous, with the exception of the recent work in [5, 6, 22]. We first review the GP framework for a set of inputs $x_1, \ldots, x_q$ that are continuous on $[0, 1]$, and then extend this framework to the mixed continuous/categorical input case.

A GP is a stochastic process (random function), $Y(\boldsymbol{x})$, over the space $\boldsymbol{x} \in \mathcal{X}$ such that for any finite set of $\boldsymbol{x}$ values, $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_k\}$,

$$\boldsymbol{Y} = [Y(\boldsymbol{x}_1), Y(\boldsymbol{x}_2), \ldots, Y(\boldsymbol{x}_k)]^T \tag{2.1}$$

has a multivariate normal distribution. Hence, a GP is completely characterized by its mean and covariance functions

$$\mu_Y(\boldsymbol{x}) = \mathrm{E}[Y(\boldsymbol{x})] \tag{2.2}$$

and

$$K_Y(\boldsymbol{x}, \boldsymbol{x}') = \mathrm{Cov}[Y(\boldsymbol{x}), Y(\boldsymbol{x}')], \tag{2.3}$$

respectively. Definitions of $\mu_Y(\boldsymbol{x})$ and $K_Y(\boldsymbol{x}, \boldsymbol{x}')$ are discussed in conjunction with Eq. (2.10) and Eqs. (2.11)-(2.15), respectively. Typically, the meta-model is then defined as

$$\hat{f}(\boldsymbol{x}) = \mathrm{E}[Y(\boldsymbol{x}) \mid Y(\boldsymbol{x}_1) = y_1, Y(\boldsymbol{x}_2) = y_2, \ldots, Y(\boldsymbol{x}_n) = y_n], \tag{2.4}$$

which is the mean of $Y(\boldsymbol{x})$ given the observed values $(\boldsymbol{x}_i, y_i)$, $i = 1, \ldots, n$. The process of obtaining $\hat{f}$ is often called Kriging after Daniel Gerhardus Krige [23].

Since $Y(\boldsymbol{x})$ is Gaussian, the expression for $\hat{f}$ can be given explicitly as

$$\hat{f}(\boldsymbol{x}) = \mu_Y(\boldsymbol{x}) + \boldsymbol{\gamma}\boldsymbol{\Sigma}^{-1}(\boldsymbol{y} - \boldsymbol{\mu}), \tag{2.5}$$

where

$$\boldsymbol{\gamma} = \left[K_Y(\boldsymbol{x}_1, \boldsymbol{x}), K_Y(\boldsymbol{x}_2, \boldsymbol{x}), \ldots, K_Y(\boldsymbol{x}_n, \boldsymbol{x})\right], \tag{2.6}$$

$$\boldsymbol{\Sigma} = \begin{bmatrix} K_Y(\boldsymbol{x}_1, \boldsymbol{x}_1) & K_Y(\boldsymbol{x}_1, \boldsymbol{x}_2) & \cdots & K_Y(\boldsymbol{x}_1, \boldsymbol{x}_n) \\ K_Y(\boldsymbol{x}_2, \boldsymbol{x}_1) & K_Y(\boldsymbol{x}_2, \boldsymbol{x}_2) & \cdots & K_Y(\boldsymbol{x}_2, \boldsymbol{x}_n) \\ \vdots & \vdots & \ddots & \vdots \\ K_Y(\boldsymbol{x}_n, \boldsymbol{x}_1) & K_Y(\boldsymbol{x}_n, \boldsymbol{x}_2) & \cdots & K_Y(\boldsymbol{x}_n, \boldsymbol{x}_n) \end{bmatrix}, \tag{2.7}$$

$$\boldsymbol{y} = [y_1, y_2, \ldots, y_n]^T, \tag{2.8}$$

$$\boldsymbol{\mu} = [\mu_Y(\boldsymbol{x}_1), \mu_Y(\boldsymbol{x}_2) \ldots, \mu_Y(\boldsymbol{x}_n)]^T, \tag{2.9}$$

and the $(\boldsymbol{x}_i, y_i)$, $i = 1, \ldots, n$ are the previously indicated observed values (pp. 160-161 [24]).

It is possible to assume a constant mean GP and let any trend in the output be accounted for as part of the random process. It is also common, however, to assume that the mean function is linear in the individual x_j. That is,

$$\mu_Y(\boldsymbol{x}) = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p, \tag{2.10}$$

where the β_j are unknown parameters that need to be estimated from the data.

There are many possible covariance structures one can use; see Section 2 of [25] for a discussion. Here we focus on one very popular class of covariances, the powered exponential family [7, 21]. With the assumption all components of $\boldsymbol{x}$ are continuous, this covariance is given by

$$K_Y(\boldsymbol{x}, \boldsymbol{x}') = \tau^2 \exp\left\{ -\sum_{j=1}^{p} \eta_j |x_j - x_j'|^{\rho_j} \right\}, \tag{2.11}$$

where $\tau^2 = \text{Var}[Y(\boldsymbol{x})]$ is the unconditional, constant (i.e., for all $\boldsymbol{x}$) variance of the process. The η_j, $j = 1, \ldots, p$, referred to as the range parameters, control how far correlation extends in each input direction. The power parameters $0 < \rho_j \leq 2$ control the shape of the correlation function, and thus the smoothness of the process. These may be estimated but are typically fixed at $\rho_j = 2$, resulting in an infinitely differentiable process.

Values of $0 < \rho_j < 2$ result in a continuous, but non-differentiable process. These two extremes can be somewhat unsettling, which has led others to consider the Matern family of covariances with which the user can specify the level of differentiability [25]. However, the powered exponential has more intuitive appeal in terms of understanding how distance controls correlation and remains the most commonly used covariance function for the use of GPs to approximate computer models [1, 2, 21, 26].

Categorical Predictors. The covariance function in Eq. (2.11) can be adapted to allow for categorical x_j variables as in [5]. To facilitate the discussion, we generalize our notation to the following. Assume that $x_1, \ldots, x_q$ are continuous on $[0, 1]$ as previously in this section, while $x_{q+1}, \ldots, x_p$ are unordered categorical variables. For simplicity, assume $x_j \in \{1, 2, \ldots, m_j\}$ for $j = q + 1, \ldots, p$, where the ordering of the integers representing the groups for x_j is completely arbitrary.

For the moment, consider possible covariance functions for just a single input which is categorical in nature. To construct a GP on just one categorical input we could assume the covariance function

$$K(x_j, x_j') = \tau^2 \left(I_{\{x_j = x_j'\}} + \nu I_{\{x_j \neq x_j'\}} \right), \tag{2.12}$$

for $0 \le \nu < 1$, where (i) $x_j, x_j' \in \{1, 2, \ldots, m_j\}$ are possible values for the categorical input, (ii) I_A is the indicator function defined by $I_A = 1$ if A and 0 otherwise. Equivalently, we could reparameterize Eq. (2.12) as $K(x_j, x_j') = \tau^2 \exp[-\eta_j I_{\{x_j \neq x_j'\}}]$. Now by combining the covariance in Eq. (2.11) for continuous inputs and that in Eq. (2.12) for categorical inputs to produce a separable covariance (i.e., product of one dimensional covariances) for $\boldsymbol{x}$ we have

$$K_Y(\boldsymbol{x}, \boldsymbol{x}') = \tau^2 \exp\left\{ -\sum_{j=1}^{q} \eta_j |x_j - x_j'|^{\rho_j} - \sum_{j=q+1}^{p} \eta_j I_{\{x_j \neq x_j'\}} \right\}. \tag{2.13}$$

The form in Eq. (2.13) assumes a covariance for categorical inputs that is *isotropic* in nature (i.e., all distinct categories have equal covariance a-priori with each other), which is reasonable in many cases. A more flexible modeling approach is also provided in [6].

Often, it is also useful to allow the observations to have an independent and identically distributed (*iid*) error term as in the traditional regression models. In this case, assume

$$Y(\boldsymbol{x}) = Z(\boldsymbol{x}) + \varepsilon, \tag{2.14}$$

where (i) $Z(\boldsymbol{x})$ is a GP with mean and covariance function μ_Z and K_Z, respectively, (ii) $\varepsilon \sim N(0, \sigma^2)$ and is independent for all values of $\boldsymbol{x}$, and (iii) $Z(\boldsymbol{x})$ independent of ε. Since the noise process ε and the actual process $Z(\boldsymbol{x})$ are assumed independent, the covariance function for $Y(\boldsymbol{x})$ is obtained by adding what is called the "nugget" term to the covariance function K. That is,

$$K_Y(\boldsymbol{x}, \boldsymbol{x}') = K_Z(\boldsymbol{x}, \boldsymbol{x}') + \sigma^2 I_{\{\boldsymbol{x}=\boldsymbol{x}'\}}, \tag{2.15}$$

where σ^2 is the variance of the *iid* errors and $I_{\{\boldsymbol{x}=\boldsymbol{x}'\}}$ is the indicator function that equals 1 if $\boldsymbol{x} = \boldsymbol{x}'$ and 0 otherwise. The term $\sigma^2 I_{\{\boldsymbol{x}=\boldsymbol{x}'\}}$ is commonly referred to as the nugget term. The term nugget is borrowed from gold mining to describe an independent source of variability much in the way that gold nuggets tend to be randomly scattered within a mine. Estimation of the model parameters β_j, η_j, τ^2, and σ^2 commonly proceeds via maximum likelihood estimation (MLE); see [27, 2, 28] for details.

The GP can be conveniently considered as a Bayesian approach to the function estimation problem. This is because the GP that we use represents our prior belief of what our computer model output will produce. Before the output is evaluated at the design points, these output values are unknown to us. However, we may have some preconceived notion about what the output will look like (e.g., we may believe that the underlying output function is "smooth" in some sense). The GP model represents this subjective uncertainty about what the output might look like. One benefit to this Bayesian framework is the relative ease of quantifying uncertainty by forming Bayesian *credible sets* for quantities of interest such as those described in Section 3. A $100(1-\alpha)\%$ Bayesian credible set for a parameter θ is defined to be a set $\mathcal{A}$ (not necessarily unique) for which

$$\Pr(\theta \in \mathcal{A} \mid \boldsymbol{Y} = \boldsymbol{y}) = 1 - \alpha, \tag{2.16}$$

see [29]. If the free parameters of the model (β_j's, τ, σ^2 ,η_j's) are estimated via MLE or by some other means and then treated as fixed, then this procedure is referred to as an empirical Bayesian approach. It is well known, however, that this approach can underestimate the uncertainty in parameter values or predictions [30]. Hence, when using this approach (i.e., GP with MLE estimation) to perform SA and UA as discussed in Section 3, we still recommend using the kriging estimate, $\hat{f}$, in conjunction with a bootstrapping procedure to produce CIs for any quantities of interest; see [4] for more details.

Because of the inherent Bayesian nature of the GP approach, it is also becoming a common practice to put *hyper*-priors on the parameters in the mean and covariance functions to make the procedure fully Bayesian (i.e., no MLE estimation involved) and approximate the posterior distribution of all of the unknown parameters with MCMC [26, 7, 14]. When the estimation is fully Bayesian, the credible sets discussed in the preceding paragraph then become a more natural way to represent the uncertainty in the estimate. This is the approach that is taken in Section 2.3, which combines GP modeling with the function ANOVA concepts described next in Section 2.2.

2.2 Adaptive COmponent Selection and Shrinkage Operator

We now review the Adaptive COmponent Selection and Shrinkage Operator (ACOSSO) surface approximation procedure [18], and extend this method to allow for categorical inputs. ACOSSO was developed under the Smoothing Spline Analysis of Variance (SS-ANOVA) modeling framework. As it is a smoothing type method, ACOSSO works best when the underlying function is somewhat smooth. For functions which are known to have sharp changes or peaks, etc., other methods are more appropriate [31, 32, 33]. ACOSSO can also work well when there are a large number of input variables. To facilitate the description of ACOSSO, we first review the univariate smoothing spline. We then describe the tensor product spline which underlies the SS-ANOVA framework under the assumption that the predictors are continuous. Lastly, we introduce the treatment of categorical predictors and the ACOSSO estimator.

Univariate Smoothing Splines. With some abuse of notation, let x_i, $i = 1, \ldots, n$, denote the i^{th} observation of a univariate input x for the discussion of univariate smoothing splines only. Without loss of generality, we restrict attention to the domain $[0, 1]$. We can always rescale the input x to this domain via a transformation. Assume that the unknown function f to be estimated belongs to the second order Sobolev space $\mathcal{S}^2 = \{g : g, g' \text{ are absolutely continuous and } g'' \in \mathcal{L}^2[0, 1]\}$. The smoothing spline estimate is given by the element $f \in \mathcal{S}^2$ that minimizes

$$\frac{1}{n}\sum_{i=1}^{n}\left[y_i - f(x_i)\right]^2 + \lambda\int_0^1 \left[f''(x)\right]^2 dx. \tag{2.17}$$

The penalty term on the right of Eq.(2.17) is an overall measure of the magnitude of the curvature (roughness) of the function over the domain. Thus, the tuning parameter λ controls the trade-off in the resulting estimate between smoothness and fidelity to the data, and thus serves the same purpose as the variance and range parameters in a GP model. Large values of λ result in smoother functions while smaller values of λ result in rougher functions that more closely match the data. Generally, λ is chosen by generalized cross validation (GCV) [34], m-fold CV [35], or related methods (pp. 239-243, [36]; pp. 42-52, [37]). The minimizer of Eq. (2.17) is technically called the cubic smoothing spline because the solution can be shown to be a natural cubic spline with knots at all of the distinct values of x_i, $i = 1, \ldots, n$ (p. 230, [36]).

Multivariate Smoothing Splines. The simplest extension of smoothing splines to multiple inputs is the additive model [37]. For instance, assume that

$$f \in \mathcal{F}_{add} = \{g : g(\boldsymbol{x}) = \sum_{j=1}^{p} g_j(x_j), g_j \in \mathcal{S}^2\}, \tag{2.18}$$

i.e., $f(\boldsymbol{x}) = \sum_{j=1}^{p} f_j(x_j)$ is a sum of univariate functions belonging to $\mathcal{S}^2$. The additive smoothing spline is the minimizer of

$$\frac{1}{n}\sum_{i=1}^{n} \left[y_i - f(\boldsymbol{x}_i)\right]^2 + \sum_{j=1}^{p} \lambda_j \int_0^1 \left[f_j''(x_j)\right]^2 dx_j \tag{2.19}$$

over $f \in \mathcal{F}_{add}$. The minimizer of the expression in Eq. (2.19), $\hat{f}(\boldsymbol{x}) = \sum_{j=1}^{p} \hat{f}_j(x_j)$, takes the form of a natural cubic spline for each of the functional components $\hat{f}_j$. Notice that there are p tuning parameters for the additive smoothing spline. These are generally determined by minimizing GCV score with respect to the λ_js.

To generalize the additive model to allow for two way interactions, we will assume f belongs to the space

$$\mathcal{F}_{2way} = \{g : g(\boldsymbol{x}) = \sum_{j=1}^{p} \sum_{k=j+1}^{p} g_{j,k}(x_j, x_k) : g_{j,k} \in \mathcal{S}^2 \otimes \mathcal{S}^2\}, \tag{2.20}$$

where $\otimes$ represents the tensor product (pp. 30-31, [38]). For two function spaces $\mathcal{G}$ and $\mathcal{H}$, the tensor product space is the vector space generated by pairwise products of functions in $\mathcal{G}$ and $\mathcal{H}$, respectively, i.e.,

$$\mathcal{G} \otimes \mathcal{H} = \left\{ \sum_{k=1}^{N} g_k h_k : g_k \in \mathcal{G}, h_k \in \mathcal{H}, N = 1, 2, \ldots \right\}. \tag{2.21}$$

An intuitive way to think of $f_{j,k} \in \mathcal{S}^2 \otimes \mathcal{S}^2$ is that for any fixed value of x, $f_{j,k}(x, \cdot) \in \mathcal{S}^2$ and $f_{j,k}(\cdot, x) \in \mathcal{S}^2$. For a complete treatment of tensor product splines and SS-ANOVA, see [39, 40, 41].

We will also need some additional notation to completely specify all of the functional components (main effects and two-way interactions) of f. Let

$$\overline{\mathcal{S}^2} = \{g \in \mathcal{S}^2 : \int_0^1 g(x)dx = 0\}. \tag{2.22}$$

This implies that $f \in \overline{\mathcal{S}^2} \otimes \overline{\mathcal{S}^2}$ has the property that

$$\int_0^1 f(x_1, x_2)dx_1 = \int_0^1 f(x_1, x_2)dx_2 = 0. \tag{2.23}$$

Now, any function $f \in \mathcal{F}_{2way}$ can be written

$$f(\boldsymbol{x}) = b_0 + \sum_{j=1}^{p} f_j(x_j) + \sum_{j=1}^{p} \sum_{k=j+1}^{p} f_{j,k}(x_j, x_k), \tag{2.24}$$

where b_0 is a constant, $f_j \in \overline{\mathcal{S}^2}$ and $f_{j,k} \in \overline{\mathcal{S}^2} \otimes \overline{\mathcal{S}^2}$. The representation in Eq. (2.24) is the functional ANOVA decomposition of f. The constant $b_0 = \int_{[0,1]^p} f(\boldsymbol{x})d\boldsymbol{x}$ can be interpreted as the overall "average" value of the function; however, technically $b_0 = \mathrm{E}[f(\boldsymbol{x})]$ only when $\boldsymbol{x}$ has a uniform distribution over $[0, 1]^p$. Also, since

$$\int_0^1 f_{j,k}(x_j, x_k)dx_j = \int_0^1 f_{j,k}(x_j, x_k)dx_k = 0, \tag{2.25}$$

the function f_j is the main effect function for variable x_j in the sense that

$$f_j(x_j) = \left(\int_{[0,1]^{p-1}} f(\boldsymbol{x}) d\boldsymbol{x}_{(-j)} \right) - b_0, \tag{2.26}$$

where $d\boldsymbol{x}_{(-j)} = dx_1, \ldots, dx_{j-1}, dx_{j+1}, \ldots, dx_p$. Additional background on the preceding relationships is given in [39] and [41].

The two-way interaction smoothing spline is given by the element $f \in \mathcal{F}_{2way}$ that minimizes

$$\frac{1}{n}\sum_{i=1}^{n}[y_i - f(\boldsymbol{x}_i)]^2 + \sum_{j=1}^{p} \lambda_j \int_0^1 \left[\frac{\partial^2}{\partial x_j^2} f_j(x_j) \right]^2 dx_j + \sum_{j=1}^{p} \sum_{k=j+1}^{p} \lambda_{j,k} \int_0^1 \int_0^1 \left[\frac{\partial^4}{\partial x_j^2 x_k^2} f_{j,k}(x_j, x_k) \right]^2 dx_j dx_k, \tag{2.27}$$

where the λ_j, $\lambda_{j,k}$ are tuning parameters that are usually chosen via cross validation. This penalizes the main effect functions exactly the same as before, and also penalizes the two-way interaction functions by a measure of roughness based on a mixed 4^{th} derivative. The minimizer of the expression in Eq. (2.27) can be obtained via matrix algebra using results from reproducing kernel Hilbert space (RKHS) theory; for details see [39], [41]. Notice that this is slightly different from the penalty for the thin plate spline ([39], [40]), which is popular in spatial statistics.

Categorical Predictors. A large advantage to the SS-ANOVA framework is the ability to handle categorical predictors with little additional complication. Assume once again that $x_1, \ldots, x_q$ are continuous on $[0, 1]$, while $x_{q+1}, \ldots, x_p$ are unordered categorical variables with $x_j \in \{1, 2, \ldots, m_j\}$ for $j = q+1, \ldots, p$. For notational convenience, let $\mathcal{G}_j = \mathcal{S}^2$ for $j = 1, \ldots, q$. Also, let the set of all functions on the domain of x_j (i.e., $\{1, 2, \ldots, m_j\}$) be denoted as $\mathcal{G}_j$ for $j = q + 1, \ldots, p$.

We can once again consider the class of additive functions

$$\mathcal{F}_{add} = \{g : g(\boldsymbol{x}) = \sum_{j=1}^{p} g_j(x_j),\ g_j \in \mathcal{G}_j\}, \tag{2.28}$$

The additive smoothing spline is then the minimizer of

$$\frac{1}{n}\sum_{i=1}^{n}\left[y_i - f(\boldsymbol{x}_i)\right]^2 + \sum_{j=1}^{q}\lambda_j \int_0^1 \left[f_j''(x_j)\right]^2 dx_j \tag{2.29}$$

over $f \in \mathcal{F}_{add}$. Notice that in the traditional setup above, there is no penalty term for roughness on the functions of the categorical predictors. To generalize to the class of functions with two way interactions, assume f belongs to the space

$$\mathcal{F}_{2way} = \left\{ g : g(\boldsymbol{x}) = \sum_{j=1}^{p}\sum_{k=j+1}^{p} g_{j,k}(x_j, x_k) \text{ for } g_{j,k} \in \mathcal{G}_j \otimes \mathcal{G}_k \right\}. \tag{2.30}$$

For $f \in \mathcal{F}_{add}$, the function $f_{j,k}$ for x_j continuous and x_k categorical amounts to allowing for a separate curve $f_{x_k}(x_j)$ for each possible value of $x_k = 1, \ldots, m_k$. We can further write f in the functional ANOVA form as in Eq. (2.24) by introducing the following notation:

$$\overline{\mathcal{G}_j} = \begin{cases} \{g : g \in \mathcal{G}_j \text{ and } \int_0^1 g(x)dx = 0\} & \text{for } j = 1, \ldots, q \\ \{g : g \in \mathcal{G}_j \text{ and } \sum_{x=1}^{m_j} g(x) = 0\} & \text{for } j = q+1, \ldots, p \end{cases} \tag{2.31}$$

Now in a similar manner to Eq. (2.24), any function $f \in \mathcal{F}_{2way}$ can be written

$$f(\boldsymbol{x}) = b_0 + \sum_{j=1}^{p} f_j(x_j) + \sum_{j=1}^{p}\sum_{k=j+1}^{p} f_{j,k}(x_j, x_k), \tag{2.32}$$

where b_0 is a constant, $f_j \in \overline{\mathcal{G}_j}$ and $f_{j,k} \in \overline{\mathcal{G}_j} \otimes \overline{\mathcal{G}_k}$.

The two-way interaction smoothing spline with categorical predictors is then given by the element $f \in \mathcal{F}_{2way}$ that minimizes

$$\frac{1}{n}\sum_{i=1}^{n}\left[y_i - f(\boldsymbol{x}_i)\right]^2 + \sum_{j=1}^{q}\lambda_j \int_0^1 \left[\frac{\partial^2}{\partial x_j^2} f_j(x_j)\right]^2 dx_j + \sum_{j=1}^{q}\sum_{k=j+1}^{q}\lambda_{j,k}\int_0^1\int_0^1 \left[\frac{\partial^4}{\partial x_j^2 x_k^2} f_{j,k}(x_j, x_k)\right]^2 dx_j dx_k$$

$$+\sum_{j=1}^{q}\sum_{k=q+1}^{p}\lambda_{j,k}\sum_{x_k=1}^{m_k}\int_0^1\left[\frac{\partial^2}{\partial x_j^2}f_{j,k}(x_j,x_k)\right]^2 dx_j, \tag{2.33}$$

where the λ_j, $\lambda_{j,k}$ are tuning parameters that are usually chosen via cross validation. The penalty in Eq. (2.33) is similar to that for the two-way interaction smoothing spline without categorical variables. However, there is no penalty on the main effect of the categorical variables. The last term in Eq. (2.33) represents a roughness penalty on the curve as a function of the continuous input x_j that results in each category of the categorical input x_k. The penalties in Eqs. (2.27) and (2.33) are partially shaped by the the computational convenience involved by using the RKHS approach. Again, the reader is referred to [39] or [41] for a detailed treatment of the computational details.

Generalizing to the ACOSSO estimate. The COmponent Selection and Shrinkage Operator (COSSO) [42] penalizes on the sum of the norms instead of the squared norms as in Eqs. (2.29) and (2.33). This modification leads to input variable selection and provides better performance than the traditional smoothing spline on data sets with a large number of predictors. For ease of presentation, we will restrict attention to the additive model for the remainder of this section. However, all of the following discussion applies directly to the two-way interaction model as well.

The penalty used to obtain the COSSO estimate is new, and is motivated by the need to perform function estimation and variable selection on categorical predictors, which are unpenalized by the traditional smoothing spline. The additive COSSO estimate, $\hat{f}(\boldsymbol{x}) = \sum \hat{f}_j(x_j)$, is given by the function $f \in \mathcal{F}_{add}$ that minimizes

$$\frac{1}{n}\sum_{i=1}^{n}[y_i - f(\boldsymbol{x}_i)]^2+\lambda\left(\sum_{j=1}^{q}\left\{\left[\int_0^1 f_j'(x_j)dx_j\right]^2+\int_0^1\left[f_j''(x_j)\right]^2 dx_j\right\}^{1/2}+\sum_{j=q+1}^{p}\left\{\sum_{x_j=1}^{m_j} f_j^2(x_j)\right\}^{1/2}\right). \tag{2.34}$$

There are four key differences in the penalty term in Eq. (2.34) relative to the additive smoothing spline of Eq. (2.29). First, there is an additional term $\left[\int_0^1 f_j'(x_j)dx_j\right]^2$, which can also be written $[f_j(1) - f_j(0)]^2$, that penalizes the magnitude of the overall trend of the functional components f_j that correspond to continuous predictors. Second, there is now a penalty on the $\mathcal{L}^2$ norms of the f_j that correspond to the categorical predictors. Third, in contrast to the squared semi-norm in the additive smoothing spline, each term in the sum in the penalty in Eq. (2.34) can be thought of as a norm over functions $f_j \in \overline{\mathcal{G}_j}$ for $j = 1, \ldots, p$. This has a similar effect to the Least Absolute Selection and Shrinkage Operator (LASSO) [43] for linear models in that it encourages some of the terms in the sum to be exactly zero. These terms are norms over the f_j; when such zeros result, some of the $\hat{f}_j$ in the solution are set to exactly zero, thus providing automatic model selection in a manner similar to the grouped LASSO [44]. Fourth, the COSSO penalty only has one tuning parameter, which can be chosen via GCV or similar means. It can be demonstrated analytically that the COSSO penalty with one tuning parameter (i.e., λ) gives as much flexibility as the smoothing spline penalty with p tuning parameters [42].

Finally, ACOSSO is a weighted version of COSSO, where a rescaled norm is used as the penalty for each of the functional components. This allows for more smoothing of irrelevant curves and more fidelity to the data for important curves. Specifically, we select as our estimate the function $f \in \mathcal{F}_{add}$ that minimizes

$$\frac{1}{n}\sum_{i=1}^{n}[y_i - f(\boldsymbol{x}_i)]^2 + \lambda_1 \sum_{j=1}^{q} w_j \left\{ \left[\int_0^1 f_j'(x_j)dx_j\right]^2 + \int_0^1 \left[f_j''(x_j)\right]^2 dx_j \right\}^{1/2} + \lambda_2 \sum_{j=q+1}^{p} w_j \left\{ \sum_{x_j=1}^{m_j} f_j^2(x_j) \right\}^{1/2}, \tag{2.35}$$

where the w_j, $0 < w_j \leq \infty$, are weights that can depend on an initial estimate of f which we denote $\tilde{f}$. Here we have two tuning parameters: one for the penalty on functional components related continuous inputs, and one for the penalty on functional components related categorical inputs. In our experience, this gives superior performance to the one tuning parameter approach when there are both continuous and categorical inputs. Our implementation of ACOSSO takes $\tilde{f}$ to be the traditional smoothing spline of Eq. (2.19), which is chosen by the GCV criterion with all $\lambda_j = \lambda$. We then use

$$w_j = \begin{cases} \left(\left[\int_0^1 \tilde{f}_j'(x_j) dx_j \right]^2 + \int_0^1 \left[\tilde{f}_j''(x_j) \right]^2 dx_j \right)^{-1} & \text{for } j = 1, \ldots, q \\ \left(\sum_{x_j=1}^{m_j} f_j^2(x_j) \right)^{-1} & \text{for } j = q+1, \ldots, p, \end{cases} \tag{2.36}$$

with λ_1 and λ_2 selected via 5-fold CV [35]. The w_j allow for more flexible estimation (less penalty) on the functional components that show more signal in the initial estimate. As shown in [18] (using only continuous inputs), this approach improves performance considerably over COSSO on many test cases and has more favorable asymptotic properties than COSSO.

The minimizer of the expression in Eq. (2.35) is obtained using an iterative algorithm and a RKHS framework similar to that used to find the minimizer of Eqs. (2.29) and (2.33) in [39, 41]. The optimization problem for the two-way interaction model can be posed in a similar way to Eq. (2.35); see [18] for more details on this and the computation of the solution. The two-way interaction model is used in the results of Sections 4 and 5. The novelty in what is presented here over [18] is the extension to a treatment of categorical inputs.

2.3 Bayesian Smoothing Spline ANOVA

A potentially problematic issue that GPs face with complex computer models is the large number of input variables present in many analyses. There have been some recent approaches to variable selection for GPs [26, 45] to alleviate this issue. The Bayesian Smoothing Spline ANOVA (BSS-ANOVA) of [19] is a GP model with a nonconventional covariance function that tackles the high dimensionality issue on two fronts: (i) variable selection to eliminate uninformative variables from the model and (ii) restricting the level of interactions involved among the variables in the model. This is done through a fully Bayesian approach which can also allow for categorical input variables with relative ease.

Once again, we first describe the BSS-ANOVA procedure under the assumption that the predictors are continuous, then introduce the treatment of categorical predictors and the BSS-ANOVA estimator.

Similar to the ACOSSO described in Section 2.2, the BSS-ANOVA model decomposes the regression function as

$$f(\boldsymbol{x}) = b_0 + \sum_{j=1}^{p} f_j(x_j) + \sum_{j=1}^{p} \sum_{k=j+1}^{p} f_{j,k}(x_j, x_k), \tag{2.37}$$

where $b_0 \in \mathcal{R}$ is the overall mean, $f_j \in \overline{\mathcal{S}^1}$ is the functional main effect for x_j, and $f_{j,k} \in \overline{\mathcal{S}^1} \otimes \overline{\mathcal{S}^1}$ is the functional interaction between x_j and x_k, and

$$\overline{\mathcal{S}^1} = \{g : g \text{ absolutely continuous and } g' \in \mathcal{L}^2[0,1], \int_0^1 g(x)dx = 0\}. \tag{2.38}$$

However, rather than estimating f by minimizing an objective function like ACOSSO, a Bayesian analysis specifies priors for f_j and $f_{j,k}$ and then analyzes their posterior distributions. The functions f_j and $f_{j,k}$ have Gaussian process priors, restricted to the appropriate spaces $f_j \in \overline{\mathcal{S}^1}$ and $f_{j,k} \in \overline{\mathcal{S}^1} \otimes \overline{\mathcal{S}^1}$, respectively, so that the posterior probability that these functions reside in the desired space is one.

To ensure that each draw (i.e., random realization) from f_j's prior (and thus posterior) is a member of $\overline{\mathcal{S}^1}$, we select a Gaussian process prior with mean zero and $\text{cov}(f_j(x_j), f_j(x_j')) = \omega_j^2 K_1(x_j, x_j')$, where ω_j^2 controls the variance of the function. Wahba [39] shows that by defining the covariance using the specialized covariance

$$K_1(x_j, x_j') = c\left[B_1(x_j)B_1(x_j') + B_2(x_j)B_2(x_j')\right] - B_4(|x_j - x_j'|)/4!, \tag{2.39}$$

where B_l is the l^{th} Bernoulli polynomial, then each draw from this Gaussian process resides in $\overline{\mathcal{S}^1}$. See [19] for a more general form of the covariance function in Eq.(2.39), where the support (i.e., any draw) of the resulting Gaussian process can be made to belong to a given m^{th} order Sobolev space for $m = 1, 2, \ldots$. Reich et.al. [19] recommend using $m = 1$ based on empirical performance; thus the $m = 1$ case is presented here.

In this covariance,

$$K_P(x_j, x_j') = c[B_1(x_j)B_1(x_j') + B_2(x_j)B_2(x_j')] \tag{2.40}$$

controls the variability of the quadratic trend and

$$K_N(x_j, x_j') = -B_4(|x_j - x_j'|)/4! \tag{2.41}$$

is the stationary covariance of the deviation from the quadratic trend. So the model essentially fits a quadratic response surface regression plus a remainder term which is a zero-mean stationary Gaussian process constrained to be orthogonal to the quadratic trend [19]. The user-specified constant c is set to 100 to give vague, yet proper, priors for the quadratic trend.

The interaction terms are also modeled as Gaussian processes. Let $f_{j,k}$ have mean zero and covariance

$$\text{Cov}[f_{j,k}(x_j, x_k), f_{j,k}(x_j', x_k')] = \omega_{j,k}^2 K_2(x_j, x_k, x_j', x_k'), \tag{2.42}$$

where

$$K_2(x_j, x_k, x_j', x_k') = \left[K_P(x_j, x_j') + K_N(x_j, x_j')\right]\left[K_P(x_k, x_k') + K_N(x_k, x_k')\right] + (c-1)K_P(x_j, x_j')K_P(x_k, x_k'). \tag{2.43}$$

For large c, the final term $(c-1)K_P(x_j, x_j')K_P(x_k, x_k')$ gives a vague prior to the bivariate quadratic trend.

Categorical Predictors. The BSS-ANOVA framework is also amenable to unordered categorical predictors. Assume $x_j \in \{1, 2, ..., m_j\}$ is categorical and let $\theta_{jl} = f_j(l)$. That is, for categorical predictors the SS-ANOVA model reduces to the usual ANOVA model where θ_{jl} is the effect for factor level l. A Bayesian ANOVA model includes priors for the θ_{jl}. To keep with the functional ANOVA construction and identify the intercept, we enforce the sum-to-zero constraint $\sum_{l=1}^{m_j} \theta_{jl} = 0$. This constraint can be achieved by defining the model $\theta_{jl}^* \overset{iid}{\sim} \mathrm{N}(0, \omega_j^2)$, $l = 1, ..., m_j$, then conditioning on the event $\sum_{l=1}^{m_j} \theta_{jl} = 0$. Since a Gaussian random vector conditional on the value of a linear combination of that vector is still Gaussian (see [24] for example), this constrained model can be equivalently expressed by calculating the covariance of the θ_{jl} conditional on $\sum_{l=1}^{m_j} \theta_{jl} = 0$. This results in the mean-zero Gaussian process with singular covariance $\mathrm{cov}(f_j(s), f_j(t)) = \omega_j^2 K_1(s, t)$, where the covariance is defined as,

$$K_1(s,t) = \frac{m_j - 1}{m_j} I(s = t) - \frac{1}{m_j} I(s \neq t), \tag{2.44}$$

and $I(\cdot)$ is the indicator function.

Interactions including categorical predictors with the covariance given in (2.44) are handled no differently than interactions between continuous predictors. For example, assume $x_1 \in \{1, ..., m_1\}$ is categorical and $x_2 \in [0, 1]$ is continuous. The covariance-based interaction results in a model for which $f_{1,2}(x_1, x_2) = h_{x_1}(x_2)$ for some $h_{x_1} \in \overline{\mathcal{S}}^1$. That is, the effect of x_2 is different within each level of x_1. An attractive feature of this covariance is that it enforces the restrictions $\int_0^1 h_{x_1}(x_2) dx_2 = 0$ for all $x_1 \in \{1, ..., m_1\}$ and $\sum_l^{m_1} h_l(x_2) = 0$ for all $x_2 \in [0, 1]$ to identify the main effects.

Variable selection. It is common in Bayesian variable selection to represent the subset of predictors included in the model with indicator variables γ_j and $\gamma_{j,k}$, where γ_j is one if the main effect for x_j is in the model and zero otherwise, and $\gamma_{j,k}$ is one if the interaction for x_j and x_k is in the model and zero otherwise. Since all functions have mean zero, if the standard deviation ω_j (or $\omega_{j,k}$) is also zero, then the function f_j (or $f_{j,k}$) is equal to zero for all values of the covariate and is effectively removed from the model. Therefore, to perform variable selection in the nonparametric setting, we specify below priors for the standard deviations ω_j and $\omega_{j,k}$ in terms of Bernoulli random variables γ_j and $\gamma_{j,k}$ to give priors with positive prior probability at zero. To account for the scale of the response via its standard deviation σ, we model the standard deviations as $\omega_j = \sigma\tau_j$ and $\omega_{j,k} = \sigma\tau_{j,k}$. We assume $\tau_j = \gamma_j\eta_j$ and $\tau_{j,k} = \gamma_{j,k}\eta_{j,k}$, where $\gamma_j, \gamma_{j,k} \overset{iid}{\sim}$ Bern(0.5) and $\eta_j, \eta_{j,k} \overset{iid}{\sim}$ HC(ρ), where ρ is the median of the half-Cauchy prior. Also, we select a flat prior for the intercept b_0 and assume $\sigma^{-2} \sim$ Gamma(0.01,0.01).

A common measure of variable importance is the posterior inclusion probability, $P(\gamma_j = 1|\boldsymbol{y})$. Often a variable is selected if $P(\gamma_j = 1|\boldsymbol{y}) > 0.5$ [46]. Using this selection criteria, Reich et al. [19] show that taking $\rho = 2/\sqrt{n}$ gives an approximate Type I error of 0.05. To compute the posterior inclusion probabilities, we may treat the model indicators γ_j and $\gamma_{j,k}$ in the same fashion as other model parameters, as unknown parameters to be explored using the MCMC algorithm described in [19]. This is referred to as stochastic search variable selection [47, 48, 49], which has been shown to be an effective method for computing posterior model probabilities.

3 Sensitivity and Uncertainty Analysis

The uncertainty in each element of $\boldsymbol{x}$ is typically characterized by a probability distribution. Such distributions are intended to numerically capture the existing knowledge about the elements of $\boldsymbol{x}$ and are often developed through an expert review process and/or through experimentation. See [12, 13] for more on the characterization of input variable uncertainty. Variance decomposition is perhaps the most informative and intuitive means with which to conduct SA (i.e., to summarize the uncertainty in an analysis outcome $y = f(\boldsymbol{x})$ resulting from uncertainty in individual elements x_j of $\boldsymbol{x}$). This procedure uses measures such as

$$s_j = \frac{\mathrm{Var}(\mathrm{E}[f(\boldsymbol{x}) \mid x_j])}{\mathrm{Var}(f(\boldsymbol{x}))} \tag{3.1}$$

and

$$T_j = \frac{\mathrm{E}(\mathrm{Var}[f(\boldsymbol{x}) \mid \boldsymbol{x}_{(-j)}])}{\mathrm{Var}(f(\boldsymbol{x}))} = \frac{\mathrm{Var}(f(\boldsymbol{x})) - \mathrm{Var}(\mathrm{E}[f(\boldsymbol{x}) \mid \boldsymbol{x}_{(-j)}])}{\mathrm{Var}(f(\boldsymbol{x}))}, \tag{3.2}$$

to quantify this uncertainty, where $\boldsymbol{x}_{(-j)} = \{x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_p\}$. The use of these measures is reviewed in [50, 12, 51]. The quantity s_j corresponds to the proportion of the uncertainty in y that can be attributed to x_j alone, while T_j corresponds to the total uncertainty that can be attributed to x_j and its interactions with other variables. These calculations require the evaluation of p-dimensional integrals which are typically approximated via Monte Carlo sampling on the joint distribution of the elements $\boldsymbol{x}$. This is too computationally intensive to be feasible for most complex computer models, and is thus the reason for describing the meta-model approaches in the previous section.

An alternative procedure to the direct evaluation of T_j and similar measures is to use a meta-model (or surrogate) for f to perform the necessary model evaluations [21, 3, 4]. That is, first collect a modest number of runs from the computer model, fit a meta-model, then use the meta-model to perform the computationally intensive calculations. Additional uncertainty introduced by the estimation error in the meta-model can be accounted for with Bayesian credible sets or bootstrapping techniques [4]. Similarly, conducting UA usually involves the estimation of a Cumulative Distribution Function (CDF) (or inverse CDF, i.e., quantiles) with Monte Carlo procedures, which can be cost-prohibitive for expensive computer models. In this case as well, meta-models can be estimated based on a limited number of computer model runs and then used to generate the thousands of model evaluations necessary to approximate a CDF. This approach to UA and SA is illustrated in Sections 4 and 5.

4 Simulation Examples

In this section, we investigate the properties of the proposed methodology for conducting UA and SA with two test models where the actual values for y can be calculated. Specifically, we will use the following two example models which are modified from examples used in several previous studies [52, 12, 53, 54, 4] to incorporate categorical inputs. Specifically, we will use

$$
\begin{aligned}
y_1 &= f_1(\boldsymbol{x}) = \begin{cases} 3.3(x_5+0.5)^4 & \text{if } x_1 = 1 \\ 0.5(x_5+0.5)^4 & \text{if } x_1 = 2 \\ 2.0(x_5+0.5)^4 & \text{if } x_1 = 3 \\ 0.6(x_5+0.5)^4 & \text{if } x_1 = 4 \\ 0.4(x_5+0.5)^4 & \text{if } x_1 = 5 \\ 4.0(x_5+0.5)^4 & \text{if } x_1 = 6 \end{cases} && (4.1) \\
y_2 &= f_2(\boldsymbol{x}) = [5.72\phi(x_1)(x_5-0.5)]^3 + 100[\phi(x_1)-0.5][\phi(x_2)-0.5]^3, && (4.2)
\end{aligned}
$$

where

$$
\phi(x) = \begin{cases} 0.05 & \text{if } x = 1 \\ 0.95 & \text{if } x = 2 \\ 0.20 & \text{if } x = 3 \\ 0.80 & \text{if } x = 4 \\ 0.40 & \text{if } x = 5 \\ 0.60 & \text{if } x = 6 \end{cases} \qquad (4.3)
$$

Each example output has 15 input variables $x_1, \ldots, x_{15}$, although not all of the inputs have an effect on each of the outputs (i.e., there are multiple "dummy" variables). Output y_1 is a function of only x_1 and x_5, while y_2 is a function of only x_1, x_2 and x_5. It is common in the analyses of real systems that some outputs are not affected (or only negligibly affected) by some of the uncertain variables under consideration [55, 11]. Inputs x_1, x_2, x_3, x_4 are discrete uniform on the integers $\{1, 2, 3, 4, 5, 6\}$. The other 11 variables are continuous uniform on $[0, 1]$. While y_1 and y_2 are technically functions of only a few inputs, the many uninformative inputs introduce a substantial increase in estimation difficulty that is often present in real problems (e.g., see the Yucca Mountain analysis presented in Section 5).

It is assumed that the output we actually observe is subject to a *small* amount of error to mimic the numerical error present in a real application; that is, we observe $y_{i,k} = f_k(\boldsymbol{x}_i)+\varepsilon_{i,k}$, with (i) $k = 1, \ldots, 2$ indexing over the two outputs and (ii) $i = 1, \ldots, n$ indexing over observations. In these examples, we let the $\varepsilon_{i,k}$ terms be generated as *iid* $\mathcal{N}(0, 0.25)$ variables. This produces signal to noise ratios (SNRs) of 159 and 558 for y_1 and y_2, respectively, where $SNR = \text{Var}[f(\boldsymbol{x})]/Var(\varepsilon)$. These analytic models have an advantage over the real model considered in Section 5. Specifically, they are fast enough to evaluate so that it is possible to calculate with great precision any quantities we wish such as the true values for the CDF of y_k, etc.

To evaluate the various meta-models, we generate 100 random samples (realizations) of size $n = 150$ from the $\boldsymbol{x}$ distribution. For each of the 100 samples, we evaluate the the outputs y_1 and y_2 at the 150 sample points, then obtain the estimate $\hat{f}_k$, for each output $k = 1, 2$, using each of the three methods (GP, ACOSSO, BSS-ANOVA) described in Section 2. We then randomly generate $N = 1000$ new points $\boldsymbol{x}_1^*, \ldots, \boldsymbol{x}_N^*$ (the same 1000 new points are used for all 100 samples), evaluate $f_1(\boldsymbol{x}_i^*)$, $f_2(\boldsymbol{x}_i^*)$, $i = 1, \ldots, N$, and estimate the integrated squared prediction error

$$ISE_k = \frac{1}{N}\sum_{i=1}^{N}(f_k(\boldsymbol{x}_i^*) - \hat{f}_k(\boldsymbol{x}_i^*))^2, \; k = 1, 2. \tag{4.4}$$

The quantity **Pred MISE**$_k$ in Table 1 is the mean over the 100 random samples of the ISE_k. **Pred 99%**$_k$ is the mean over the 100 samples of the 99-th percentile of the squared predition error $(f_k(\boldsymbol{x}_i^*) - \hat{f}_k(\boldsymbol{x}_i^*))^2$. Finally, the sample $f_k(\boldsymbol{x}_i^*)$ is used to estimate the CDF of f_k and **CDF ISE**$_k$ is the mean over the 100 samples of the integrated squared estimation error of the true CDF curve to the estimated CDF via the meta-model values $\hat{f}_k(\boldsymbol{x}_i^*)$.

Table 1: Simulation Summary Statistics over 100 realizations of outputs y_1 and y_2.

Test Function 1

Estimator	Pred MISE[1]	Pred 99%[2]	CDF ISE[3]
ACOSSO	0.28 (0.03)	3.98 (0.60)	0.006 (0.000)
BSS-ANOVA	0.18 (0.01)	3.09 (0.46)	0.006 (0.000)
GP	1.09 (0.06)	18.26 (1.73)	0.010 (0.001)

Test Function 2

Estimator	Pred MSE	Pred 99%	CDF ISE
ACOSSO	1.61 (0.21)	21.31 (2.72)	0.011 (0.001)
BSS-ANOVA	3.81 (0.19)	48.04 (2.94)	0.015 (0.001)
GP	4.87 (0.25)	77.02 (7.90)	0.018 (0.001)

[1] mean (over 100 samples) of the integrated squared prediction error.

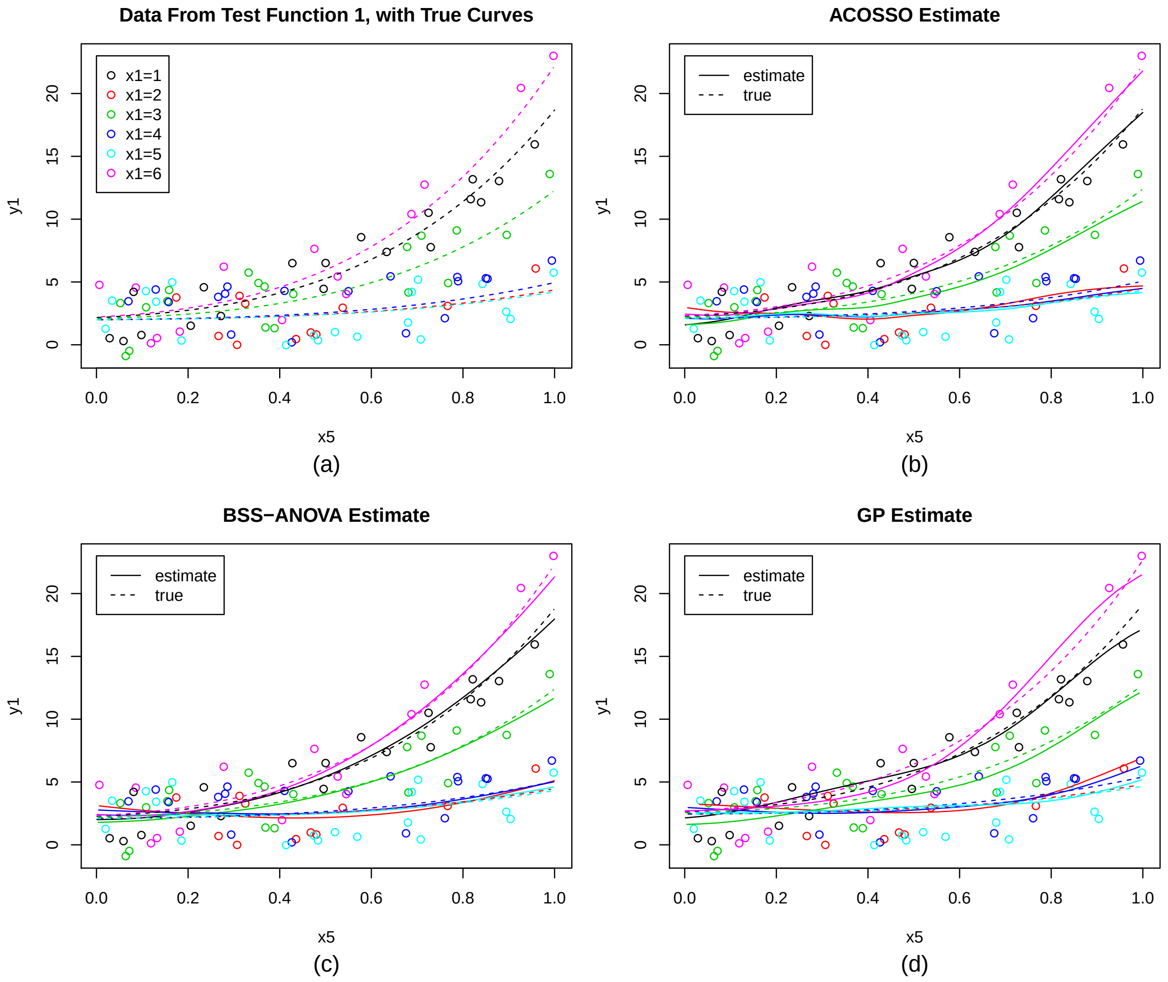
[2] mean (over 100 samples) of the 99-th percentile squared prediction error.

[3] mean (over 100 samples) of the CDF integrated squared estimation error.

Figure 1 displays the fitted meta-models for y_1 from the first sample of size $n = 150$ of the 100 samples described above. The dashed curves in Figure 1(a) are the true main effect curves for x_5 for the various levels of x_1, i.e., $\mathrm{E}(y_1 \mid x_1, x_5)$ is plotted across x_5 for each level of x_1. Figures 1(b), 1(c), 1(d) give the estimated main effects from ACOSSO, BSS-ANOVA, and GP meta-models, (described in Sections 2.2, 2.3, and 2.1) respectively, i.e., $\mathrm{E}(y_1 \mid x_1, x_5)$ is estimated via Monte Carlo of the meta-model $\hat{f}$ and plotted across x_5 for each level of x_1. From the plots it is clear that ACOSSO and BSS-ANOVA estimate the true model very well. GP also estimates the model well, but does have more estimation error than the other two methods, particularly at the higher values of x_5.

Table 1 provides the summary statistics defined above for the entire simulation of 100 samples for both outputs y_1 and y_2. BSS-ANOVA has the least estimation error for the y_1 analysis, with ACOSSO close behind, and GP has the most estimation error. For the y_2 analysis, ACOSSO is the best method, with BSS-ANOVA next then GP.

Figure 1: Output y_1 for the first random sample of size $n = 150$, (a) Scatterplot of y_1 plotted against x_5, dashed curves are the true main effect curves for x_5 conditional on the individual levels of x_1, (b) Main effect curves resulting from ACOSSO estimate, (c) Main effect curves resulting from BSS-ANOVA estimate, and (d) Main effect curves resulting from GP estimate.



(a)

(b)

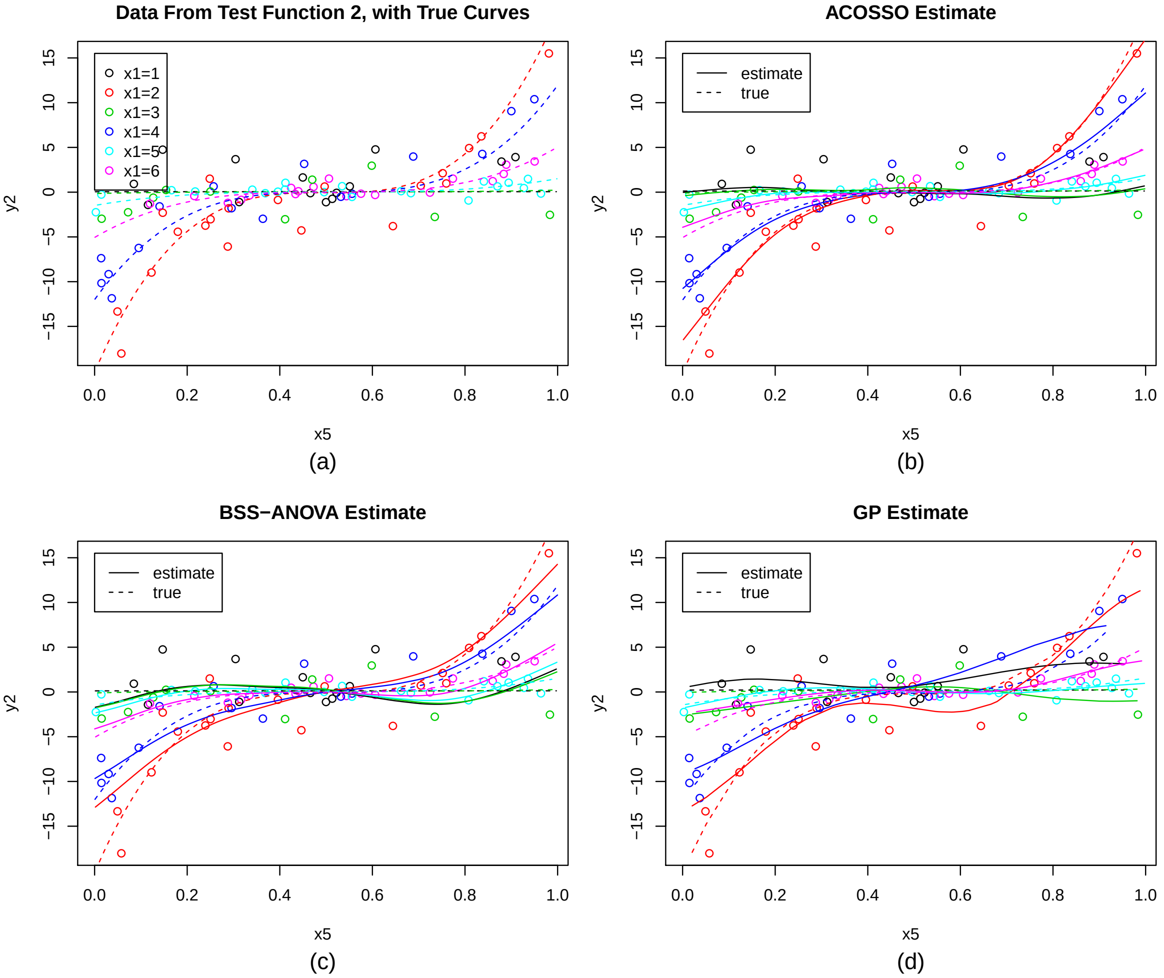
(c)

(d)

Figure 2 displays the fitted meta-models once again, this time on the output y_2 for a given data realization. In this plot it is clear that ACOSSO estimates the second output function more accurately than the other two methods. BSS-ANOVA also estimates the function well, but is not as flat as needed across x_5 for $x_1 = 1$ and $x_1 = 3$, and does not have quite enough curvature across x_5 for $x_1 = 2$. GP has the most trouble of the three methods on this output, as it appears to be undersmoothing across x_5 for $x_1 = 2$, yet oversmoothing across x_5 for $x_1 = 4$.

ACOSSO estimates are much faster to obtain than BSS-ANOVA and GP estimates,

Figure 2: Output y_2 for the first random sample of size $n = 150$, (a) Scatterplot of the output y_2 plotted against input x_5 for each possible value of the categorical input x_1, dashed curves are the true main effect curves for x_5 for the various levels of x_1, (b) Main effect curves resulting from ACOSSO estimate, (c) Main effect curves resulting from BSS-ANOVA estimate, and (d) Main effect curves resulting from GP estimate.



(a)

(b)

(c)

(d)

since they can be solved as a quadratic programming problem with linear constaints [18]. BSS-ANOVA requires an MCMC procedure [19] to obtain the estimate, and GP estimation requires an iterative Newton Raphson procedure. However, if confidence bounds for parameter estimates are desired, the BSS-ANOVA procedure has faster computation times than ACOSSO, as ACOSSO will require a bootstrap procedure [4] to produce confidence sets for parameter(s) of interest. When using the BSS-ANOVA procedure, however, confidence statements can naturally be made based on the approximate posterior distribution of the parameter(s) of interest, which can be computed with minimal additional cost after (or during) the MCMC estimation. When “confidence” sets are constructed to include say 95% of the posterior probability for a given parameter, they are typically called 95% *credible* sets, to distinguish them from a 95% *confidence* set which implies 95% coverage of the true value of the parameter under repeated sampling (see Eq.(2.16) for a formal definition of a credible set).

The GP estimates are faster to obtain than BSS-ANOVA estimates since the GP procedure used here is based off of MLEs for parameters (i.e., not MCMC). Credible sets from the GP can be produced with an empirical Bayesian approach [56], in which case it is the fastest of the methods to produce “confidence” sets. Credible sets for ACOSSO could be produced using an empirical Bayesian approach in a similar manner as well. However, the empirical Bayesian approach is known to provide overly optimistic confidence statements as it ignores parameter estimation error. To avoid this issue, [14] and [26] use a fully Bayesian approach (i.e., incorporating parameter estimation uncertainty using the posterior distribution). In a fully Bayesian approach, requiring an MCMC routine, BSS-ANOVA will be more efficient than the traditional GP for moderate to large samples, as the GP approach requires a $n \times n$ matrix solve for each MCMC iteration, while BSS-ANOVA only requires a $n \times n$ matrix solve once prior to the MCMC (see [19] for more computational details).

5 SA and UA on Results from Yucca Mountain

This example comes from a computational model for the proposed Yucca Mountain repositrory for high-level radioactive waste [16, 17]. There are a total of 145 input variables (several of which are categorical in nature) for the particular analysis situation under consideration and dozens of time dependent responses. The models involved are very expensive to run, and were evaluated at input values provided by a Latin Hypercube sample [57, 58] of size $n = 300$. A subset of the entire dataset is available at `http://www.lanl.gov/~storlie/CompModSA/ym_data.txt`. The output variable for this illustration is

ESIC239C.10K: Cumulative release of Ic (i.e. Glass) colloid of 239 Pu (Plutonium 239) out of the engineered barrier system into the unsaturated zone at 10,000 years.

The input variables that appear in plots and tables below are:

TH.INFIL: Categorical variable describing different scenarios for infiltration and thermal conductivity in the region surrounding the drifts (see p.K-14, [16]).

CPUCOLWF: Concentration of irreversibly attached plutonium on glass/waste form colloids when colloids are stable (mol/L).

RHMUNO65: The in-drift precipitated salts (IDPS) process model uncertainty factor for the logarithm of the chloride to nitrate ratio of the in-drift waters at relative humidity >65% and ≤85% (dimensionless).

FHHISSCS: Frenkel Halsey Hill water vapor adsorption isotherm parameter, s, for commercial spent nuclear fuel (CSNF) rind (dimensionless).

SEEPUNC: Uncertainty factor to account for small-scale heterogeneity in fracture permeability (dimensionless).

PH2DHLS: Pointer variable used to determine pH in the defense high level waste (DHLW) cell (Cell 1a) of codisposed spent fuel (CDSP) waste packages under liquid influx conditions (dimensionless).

PH2MCONS: Pointer variable used to determine pH in the multicanister overpack (MCO) cell (Cell 1b) of CDSP waste packages under vapor influx conditions (dimensionless).

For a more detailed description of all input and output variables for the Yucca Mountain analysis, see Tables K3-1, K3-2, K3-3, and K3-4 of [16].

In this example, we first conducted some basic data display techniques such as scatterplots and boxplots of the data to see if any transformations were needed to help spread the data out evenly over the respective ranges. This resulted in the following transformation

$$\mathit{ESIC239C.10K} = \log(\mathit{ESIC239C.10K} + 0.1).$$

We then looked at correlations among input variables, and removed any input variables that had a correlation higher than 0.95 with another input variable (only the first of such input variables was kept). This resulted in removal of four input variables. Finally, to allow for efficient computation and better estimation, we screened the remaining 141 input variables for importance to the output *ESIC239C.10K* using the SRD/RCC scatterplot test for non-randomness described in [59, 13]. We used $\alpha = 0.10$ as a somewhat liberal threshold in the SRD/RCC test in order to not exclude any inputs which could be informative. This resulted in a remaining 17 inputs (15 continuous and 2 categorical) that could possibly have a non-negligable effect on the output.

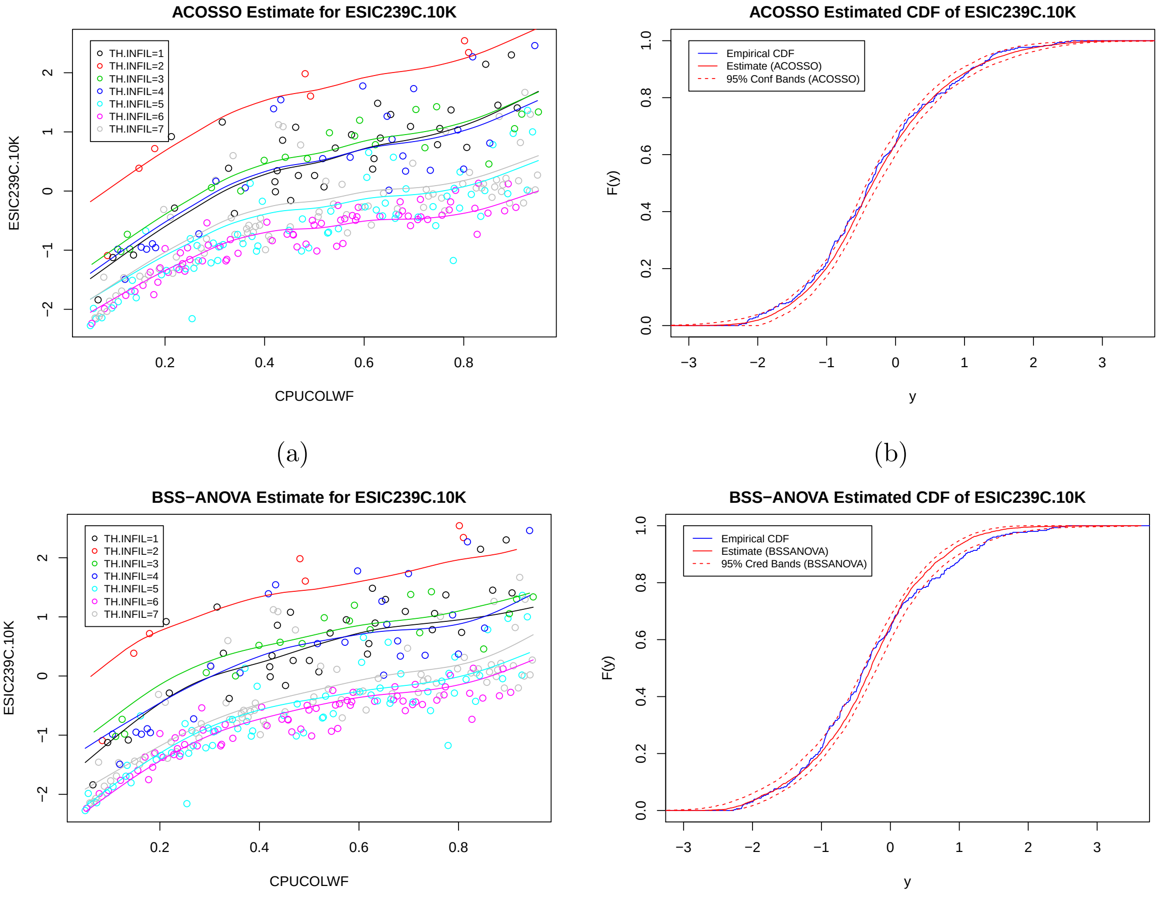
Figure 3(a) displays the main effect curves for *CPUCOLWF* at each level of the categorical variable *TH.INFIL* estimated with the ACOSSO meta-model, i.e.,

$$\mathrm{E}(\mathit{ESIC239C.10K} \mid \mathit{CPUCOLWF}, \mathit{TH.INFIL}) \tag{5.5}$$

estimated via monte carlo of the meta-model $\hat{f}$and plotted across *CPUCOLWF* for each level of *TH.INFIL*. Figure 3(c) displays the same plot using the BSS-ANOVA estimate. *CPUCOLWF* and *TH.INFIL* are the two most influential inputs on the output *ESIC239C.10K* as determined by a SA of *ESIC239C.10K*, which is presented later in this section.

Figure 3(b) displays the estimated CDF produced from the ACOSSO meta-model along with simultaneous 95% confidence bands obtained via bootstrapping [4]. Figure 3(d) displays the estimated CDF from BSS-ANOVA along with with simultaneous 95% Bayesian credible bands obtained from the approximate posterior distribution resulting from the MCMC routine described in [19]. The blue curves in these plots are the empirical (i.e., data) CDF obtained using 300 computer model evaluations. These CDFs are less smooth because there are jumps at each of the 300 points; in contrast, the ACOSSO and BSS-ANOVA estimates use 10,000 points to produce the CDFs, resulting

Figure 3: Uncertainty results for *ESIC239C.10K*: (a) *ESIC239C.10K* scatterplot with fitted main effect curves from ACOSSO, (b) Estimated CDF (solid line) for *ESIC239C.10K* from ACOSSO along with 95% simultaneous confidence bands (dashed), (c) *ESIC239C.10K* scatterplot with fitted main effect curves from BSS-ANOVA, and (d) Estimated CDF (solid line) for *ESIC239C.10K* from BSS-ANOVA along with 95% simultaneous credible bands (dashed).



(a)

(b)

(c)

(d)

in a much smoother appearance. ACOSSO and BSS-ANOVA give very similar estimation results for this example, which is reassuring with respect to the validity of the UA and SA results.

Table 2 provides the SA results for *ESIC239C.10K* obtained using ACOSSO and BSS-ANOVA. The quantity R^2 is defined as

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{f}(\boldsymbol{x}_i))^2}{\sum_{i=1}^{n}(y_i - 1/n\sum_{j=1}^{n} y_i)^2} \tag{5.6}$$

and provides a measure of what proportion of the empirical variance in the output is accounted for by the meta-model (i.e., how well the meta-model matches the data). Quantities above 0.90 and closer to 1.00 are preferred but an $R^2 = 1.00$ does not necessarily mean that the meta-model will provide good prediction for new observations. Also provided are estimates of the total variance index (T_j) for the informative inputs as defined in Eq.(3.2) via a Monte Carlo sample of the meta-model as described in [4]. The 95% Confidence Intervals (CIs) for the T_j are obtained via bootstrapping for ACOSSO [4] and Bayesian credible sets for BSS-ANOVA [19]. In our experience, it is beneficial to consider results from both frequentist and Bayesian analyses for complex problems to determine if both approaches give similar results. If this is not the case, then it is important to assess the problem further, e.g., get an expert's opinion on the results, try to understand why the results are different, and/or perform more model runs if possible.

In this example, it is clear that the main drivers of the uncertainty in the output *ESIC239C.10K* are *CPUCOLWF* and *TH.INFIL*. As shown in Table 2, *CPUCOLWF* (along with its interactions with other inputs) accounts for approximately 47% to 60% of the total uncertainty in the output. *TH.INFIL* and its interactions account for approximately 25% to 52% of the total uncertainty. *RHMUNO65*, *FHHISSCS*, and *SEEPUNC* (along with their interations) account for approximately another 3-6%, 4-5% and ∼2%, respectively, of the uncertainty. The BSS-ANOVA method identified two more inputs (*PH2DHLS* and *PH2MCONS*) than ACOSSO did with these variables (along with their interactions) potentially contributing ∼7% of the total variance.

6 Conclusions and Further Work

Two new meta-models, ACOSSO and BSS-ANOVA, are presented, which are similar in their functional ANOVA construction. However, ACOSSO is a frequentist type approach in the sense that no prior distributions are explicitly assumed; in contrast BSS-ANOVA

Table 2: Sensitivity Analysis Results for *ESIC239C.10K* obtained using ACOSSO and BSS-ANOVA.

Meta-model: ACOSSO Model $R^2 = 0.95$[1]			Meta-model: BSS-ANOVA Model $R^2 = 0.94$		
Input	$\hat{T}_j$[2]	95% T_j CI[3]	Input	$\hat{T}_j$	95% T_j CI
CPUCOLWF	0.565	(0.473, 0.621)	*CPUCOLWF*	0.524	(0.475, 0.579)
TH.INFIL	0.424	(0.360, 0.518)	*TH.INFIL*	0.304	(0.248, 0.361)
RHMUNO65	0.063	(0.052, 0.126)	*RHMUNO65*	0.026	(0.010, 0.056)
FHHISSCS	0.053	(0.041, 0.106)	*FHHISSCS*	0.035	(0.003, 0.073)
SEEPUNC	0.020	(0.000, 0.040)	*SEEPUNC*	0.020	(0.008, 0.036)
			PH2DHLS	0.067	(0.038, 0.117)
			PH2MCONS	0.065	(0.034, 0.094)

[1] proportion of output variance in the the sample that is accounted for by the meta-model as defined in Eq.(5.6).

[2] estimate of the total variance index as defined in Eq.(3.2) via a Monte Carlo sample of the meta-model.

[3] 95% Confidence Intervals (CIs) for the T_j, obtained via bootstrapping for the ACOSSO meta-model and via Bayesian credible sets for the BSS-ANOVA meta-model.

is a fully Bayesian estimation procedure. ACOSSO estimates are generally faster to obtain than BSS-ANOVA estimates, unless confidence bounds for parameter estimates are desired, in which case BSS-ANOVA is somewhat faster.

Functional ANOVA construction and variable selection (e.g., ACOSSO and BSS-ANOVA) can help to increase efficiency in function estimation and easily allows for categorical inputs as special cases. When using functional ANOVA construction, the main effect and interaction functions are immediately available (i.e., no need to numerically integrate). The practical benefits (e.g., better surface and CDF approximation) of the functional ANOVA construction, inherent in ACOSSO and BSS-ANOVA, over the traditional GP meta-model have been empirically demonstrated on two established test cases. ACOSSO and BSS-ANOVA were then used to perform a UA and SA on an output from the Yucca Mountain analysis, which had several categorical inputs, one of which was very influential.

Functional ANOVA construction also lends itself well to allowing for "nonstationarity" in function estimation [33], i.e., in situations where the output has sharp or rapid changes in some parts of its domain, but is relatively smooth elsewhere. Approaches to nonstationary modeling such as [60] and [61] are notoriously difficult to apply in a high dimensional input space. In functional ANOVA construction however, the overall function (which is potentially quite complex) is decomposed into fairly simple functions (i.e., main effects or 2-way interactions) as indicated in Eq.(2.24). Nonstationarity can be included in all or only some of these functional components. Since each of these functional components lives in a low dimensional input space, it makes the extension to nonstationarity much easier than for a general function of p inputs. This is a subject of further work

References

[1] J. Sacks, W.J. Welch, T.J. Mitchel, and H.P. Wynn. Design and analysis of computer experiments. *Statistical Science*, 4:409–435, 1989.

[2] M.C. Kennedy and A. O'Hagan. Bayesian calibration of computer models (with discussion). *Journal of the Royal Statistical Society B*, 63:425–464, 2001.

[3] C.B. Storlie and J.C. Helton. Multiple predictor smoothing methods for sensitivity analysis: Description of techniques. *Reliability Engineering and System Safety*, 93:28–54, 2008.

[4] C.B. Storlie, L.P. Swiler, J.C. Helton, and C.J. Sallaberry. Implementation and evaluation of nonparametric regression procedures for sensitivity analysis of computationally demanding models. *Reliability Engineering and System Safety*, 94:1735–1763, 2009.

[5] V.R. Joseph and J.D. Delaney. Functionally induced priors for the analysis of experiments. *Technometrics*, 49:1–11, 2007.

[6] P. Qian, H. Wu, and C.F.J. Wu. Gaussian process models for computer experiments with qualitative and quantitative factors. *Technometrics*, 50:383–396, 2008.

[7] R.B. Gramacy and H.K. Lee. Bayesian treed Gaussian process models with an application to computer modeling. *Journal of the American Statistical Association*, 103:1119–1130, 2008.

[8] D. Higdon, J. Gattiker, B. Williams, and M. Rightley. Computer model validation using high-dimensional output. *Journal of the American Statistical Association*, 103:570–583, 2008.

[9] M.J. Bayarri, J.O. Berger, J. Cafeo, G. Garcia-Donato, F. Liu, J. Palomo, R.J. Parthasarathy, R. Paulo, J. Sacks, and D. Walsh. Computer model validation with functional output. *Annals of Statistics*, 35:1874–1906, 2007.

[10] J.C. Helton and R.J. Breeding. Calculation of reactor accident safety goals. *Reliability Engineering and System Safety*, 39:129–158, 1993.

[11] J.C. Helton and M.G. Marietta, Editors. Special issue: The 1996 performance assessment for the Waste Isolation Pilot Plant. *Reliability Engineering and System Safety*, 69:1–451, 2000.

[12] A. Saltelli, Chan K., and E.M. Scott, editors. *Sensitivity Analysis*. New York, NY: Wiley, 2000.

[13] J.C. Helton, J.D. Johnson, C.J. Sallaberry, and C.B. Storlie. Survey of sampling-based methods for uncertainty and sensitivity analysis. *Reliability Engineering and System Safety*, 91:1175–1209, 2006.

[14] D. Higdon, M. Kennedy, J.C. Cavendish, J.A. Cafeo, and R.D. Ryne. Combining field data and computer simulations for calibration and prediction. *SIAM Journal on Scientific Computing*, 26:448–466, 2004.

[15] R.L. Iman and J.C. Helton. An investigation of uncertainty and sensitivity analysis techniques for computer models. *Risk Analysis*, 8:71–90, 1988.

[16] Sandia National Laboratories. Total system performance assesment model/analysis for the license application. *U.S. Department of Energy Office of Civilian Radioactive Waste Management*, MDL-WIS-PA-000005 Rev 00, AD 01.Las Vegas, NV, 2008.

[17] J.C. Helton, C.W. Hansen, and C.J. Sallaberry. Uncertainty and sensitivity analysisin performance assesment for the proposed repository for high-level radioactive waste at yucca mountain, nevada. *Reliability Engineering and System Safety* (to appear), 2012.

[18] C.B. Storlie, H.D. Bondell, B.J. Reich, and H.H. Zhang. Surface estimation, variable selection, and the nonparametric oracle property. *Statistica Sinica*, 21:679–705, 2010.

[19] B.J. Reich, C.B. Storlie, and H.D. Bondell. Variable selection in Bayesian smoothing spline ANOVA models: Application to deterministic computer codes. *Technometrics*, 51:110–120, 2009.

[20] T. Santner, B. Williams, and W. Notz. *The Design and Analysis of Computer Experiments*. New York, NY: Springer, 2003.

[21] J.E. Oakley and A. O'Hagan. Probabilistic sensitivity analysis of complex models: A Bayesian approach. *Journal of the Royal Statistical Society B*, 66:751–769, 2004.

[22] G. Han, T.J. Santner, and W.I. Notz. Prediction for computer experiments having quantitative and qualitative input variables. *Technometrics*, 51(3):278–288, 2009.

[23] G. Matheron. Principles of geostatistics. *Economic Geology*, 58:1246–1266, 1963.

[24] R.A. Johnson and D.W. Wichern. *Applied Multivariate Statistical Analysis*. Upper Saddle River, NJ: Pearson Prentice Hall, 6th edition, 2007.

[25] M.L. Stein. *Interpolation of Spatial Data*. Springer-Verlag, New York, NY, 1999.

[26] C. Linkletter, D. Bingham, N. Hengartner, D. Higdon, and K. Ye. Variable selection for Gaussian process models in computer experiments. *Technometrics*, 48:478–490, 2006.

[27] W.J. Welch, R.J. Buck, J. Sacks, H.P. Wynn, T.J. Mitchell, and M.D. Morris. Screening, predicting, and computer experiments. *Technometrics*, 34:15–25, 1992.

[28] G. Dancik. mlegp: an R package for Gaussian process modeling and sensitivity analysis. *http://cran.r-project.org/web/packages/mlegp/vignettes/gp.pdf*, 2007.

[29] A. Gelman, J.B. Carlin, H.S. Stern, and D.B. Rubin. *Bayesian Data Analysis*. Chapman & Hall, London, 2nd edition, 2004.

[30] N.A.C. Cressie. *Statistics for Spatial Data*. New York, NY: John Wiley & Sons, 1993.

[31] J.H. Friedman. Multivariate adaptive regression splines (with discussion). *Annals of Statistics*, 19:1–141, 1991.

[32] D.G.T. Denison, B.K. Mallick, and A.F.M. Smith. Bayesian MARS. *Statistics and Computing*, 8:337–346, 1998.

[33] C.B. Storlie, H.D. Bondell, and B.J. Reich. A locally adaptive penalty for estimation of functions with varying roughness. *Journal of Computational and Graphical Statistics*, 19:569–589, 2010.

[34] P. Craven and G. Wahba. Smoothing noisy data with spline functions: estimating the correct degree of smoothing by the method of generalized cross-validation. *Numerical Mathematics*, 31:377–403, 1979.

[35] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, 2:1137–1143, 1995.

[36] R.L. Eubank. *Nonparametric Regression and Spline Smoothing*. CRC Press, 1999.

[37] T. Hastie and R.J. Tibshirani. *Generalized Additive Models*. Chapman & Hall/CRC, 1990.

[38] A. Berlinet and C. Thomas-Agnan. *Reproducing Kernel Hilbert Spaces in Probability and Statistics*. Kluwer Academic Publishers, Norwell, MA, 2004.

[39] G. Wahba. *Spline Models for Observational Data*. CBMS-NSF Regional Conference Series in Applied Mathematics, 1990.

[40] M. Schimek, editor. *Smoothing and Regression: Approaches, Computation, and Application*. John Wiley, New York, 2000.

[41] C. Gu. *Smoothing Spline ANOVA Models*. Springer-Verlag, New York, 2002.

[42] Y. Lin and H. Zhang. Component selection and smoothing in smoothing spline analysis of variance models. *Annals of Statistics*, 34:2272–2297, 2006.

[43] R.J. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society B*, 58:267–288, 1996.

[44] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society B*, 68:49–67, 2006.

[45] A. Marrel, B. Iooss, F. Van Dorpe, and E. Volkova. An efficient methodology for modelling complex computer codes with Gaussian processes. *Computational Statistics and Data Analysis*, 52:4731–4744, 2008.

[46] M. Barbieri and J. Berger. Optimal predictive model selection. *Annals of Statistics*, 32:870–897, 2004.

[47] E.I. George and R.E. McCulloch. Variable selection via Gibbs sampling. *Journal of the American Statistical Association*, 88:881–889, 1993.

[48] H. Chipman. Bayesian variable selection and related predictors. *Canadian Journal of Statistics*, 24:17–36, 1996.

[49] E.I. George and R.E. McCulloch. Approaches for Bayesian variable selection. *Statistica Sinica*, 7:339–373, 1997.

[50] A. Saltelli, S. Tarantola, and Chan K. A quantitative model-independent method for global sensitivity analysis of model output. *Technometrics*, 41:39–56, 1999.

[51] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, and S. Tarantola, editors. *Global Sensitivity Analysis: The Primer*. West Sussex, England: Wiley, 2008.

[52] F. Campolongo, A. Saltelli, T. Sorensen, and S. Tarantola. Hitchhiker's guide to sensitivity analysis. *Sensitivity Analysis*, pages 15–47, 2000. Ed. A. Saltelli, K. Chan, and M. Scott. New York, NY: John Wiley & Sons.

[53] J.C. Helton and F.J. Davis. Illustration of sampling-based methods for uncertainty and sensitivity analysis. *Risk Analysis*, 22:591–622, 2002.

[54] C.B. Storlie and J.C. Helton. Multiple predictor smoothing methods for sensitivity analysis: Example results. *Reliability Engineering and System Safety*, 93:55–77, 2008.

[55] P. Vaughn, J.E. Bean, J.C. Helton, M.E. Lord, R.J. MacKinnon, and J.D. Schreiber. Representation of two-phase flow in the vicinity of the repository in the 1996 performance assessment for the Waste Isolation Pilot Plant. *Reliability Engineering and System Safety*, 69:205–226, 2000.

[56] J.E. Oakley and A. O'Hagan. Bayesian inference for the uncertainty distribution of computer model outputs. *Biometrika*, 89:769–784, 2002.

[57] M.D. McKay, Beckman R.J., and W.J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21:239–245, 1979.

[58] J.C. Helton and F.J. Davis. Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems. *Reliability Engineering and System Safety*, 81:23–69, 2003.

[59] S.C. Hora and J.C. Helton. A distribution-free test for the relationship between model input and output when using latin hypercube sampling. *Reliability Engineering and System Safety*, 79:333–339, 1996.

[60] D. Higdon, J. Swall, and J. Kern. Non-stationary spatial modeling. In J.M. Bernardo, J.O. Berger, A.P. Dawid, and A.F.M. Smith, editors, *Bayesian Statistics 6*, pages 761–768. Oxford, UK: Oxford University Press., 1999.

[61] M. Fuentes. Spectral methods for nonstationary spatial processes,. *Biometrika*, 89:197–210, 2002.